

CLAIMS

Having thus described our invention, what we claim as new, and desire to secure by Letters Patent is:

- Sub a1
1. A method for identifying to a user, the differences between elements of two hierarchically structured files, comprising the steps of:
 - comparing the elements of a base file to the elements of a modified file;
 - providing to the user a tree structure, said tree structure combining the elements of said base and said modified files; and
 - highlighting the differences between said elements of said base and said modified files.
 2. The method of claim 1 further comprising the step of allowing the user to resolve said differences between elements, thereby creating a merged file containing elements from said base file and elements from said modified file.
 3. The method of claim 2 which includes indicating to the user differences between elements by one of the identifiers: new, changed or removed.
 4. The method of claim 3 which includes, for an element identified as new, providing the user with the following options:
 - a) do not use the new element, whereby the new element is not incorporated into said merged file;
 - and

1 b) use the new element, whereby the new
2 element and children thereof, if any, are incorporated
3 into said merged file.

1 5. The method of claim 3 which includes, for an
2 element identified as changed, providing the user with
3 the following options:

4 a) use old, where conflict, whereby for the
5 merged file the changed element is taken from the base
6 file together with unresolved children thereof, if any;
7 and

8 b) use new, where conflict, whereby for the
9 merged file the changed element is taken from the
10 modified file together with unresolved children thereof,
11 if any.

1 6. The method of claims 3, which includes, for an
2 element identified as removed, providing the user with
3 the following options:

4 a) do not delete, whereby the merged file has
5 the element as it exists in the base file; and

6 b) delete from the base file, whereby the
7 merged file does not have the element that was deleted
8 from the base file.

1 7. The method of claim 1 wherein the step of
2 providing to user a tree structure comprises visually
3 displaying the tree structure.

1 8. The method of claim 7 wherein visually
2 displaying the tree structure comprises displaying to the
3 user a screen containing three panes, the first pane

1 displaying said tree structure, the second pane
2 displaying an element of said base file, and the third
3 pane displaying an element of the modified file.

1 9. The method of claim 8 which includes, when the
2 user selects an element of the tree structure displayed
3 in the first pane, displaying the source code for the
4 selected element:

- 5 a) in the second pane if the selected element
6 exists in the base file; and
7 b) in the third pane if the selected element
8 exists in the modified file.

1 10. The method of claim 1 wherein the step of
2 comparing uses an ID attribute of the elements of the
3 base file and the modified file being compared.

1 11. The method of claim 1 wherein the step of
2 comparing uses a name attribute of the elements of the
3 base file and the modified file being compared.

1 12. The method of claim 1 wherein said
2 hierarchically structured files are XML (eXtensible
3 markup language) files and wherein the step of comparing
4 uses:

5 if provided by the elements of the base and
6 modified files being compared, an attribute of type ID;

7 if an attribute of type ID is not provided by
8 the elements of the base and modified files being
9 compared, a <Uuid> tag if provided by the elements of the
10 base and modified files being compared;

1 if an attribute of type ID and a <Uuid> tag is
2 not provided by the elements of the base and modified
3 files being compared, a name attribute if provided by the
4 elements of the base and modified files being compared;
5 and

6 if an attribute of type ID, a <Uuid> tag and a
7 name attribute is not provided by the elements of the
8 base and modified files being compared, a concatenation
9 of a tag of the element and a value of the element.

1 13. The method of claim 1 wherein said
2 hierarchically structured files are XML (eXtensible
3 Markup Language) files.

1 14. A method for visually identifying to a user,
2 the differences between elements of a hierarchical base
3 data structure and a hierarchical modified data
4 structure, comprising the steps of:

5 comparing the elements of said base data
6 structure to the elements of said modified data
7 structure;

8 displaying to the user a tree structure, said
9 tree structure combining the elements of said base and
10 modified data structures; and

11 highlighting the differences between said
12 elements of said base and modified data structures.

1 15. A program storage device readable by a data
2 processing system, tangibly embodying a program of
3 instructions, executable by said data processing system
4 to perform the method steps of claim 1.

1 16. A system for identifying to a user, the
2 differences between elements of two hierarchically
3 structured files, comprising:

4 means for comparing the elements of a base file
5 to the elements of a modified file;

6 means for providing to the user a tree
7 structure, said tree structure combining the elements of
8 said base and said modified files; and

9 means for highlighting the differences between
10 said elements of said base and said modified files.

1 17. The system of claim 16 further comprising means
2 for allowing the user to resolve said differences between
3 elements, thereby creating a merged file containing
4 elements from said base file and elements from said
5 modified file.

1 18. The system of claim 17 which includes means for
2 indicating to the user differences between elements by
3 one of the identifiers: new, changed or removed.

1 19. The system of claim 18 which includes, for an
2 element identified as new, providing the user with the
3 following options:

4 a) do not use the new element, whereby the
5 new element is not incorporated into said merged file;
6 and

1 b) use the new element, whereby the new
2 element and children thereof, if any, are incorporated
3 into said merged file.

1 20. The system of claim 18 which includes, for an
2 element identified as changed, means for providing the
3 user with the following options:

4 a) use old, where conflict, whereby for the
5 merged file the changed element is taken from the base
6 file together with unresolved children thereof, if any;
7 and

8 b) use new, where conflict, whereby for the
9 merged file the changed element is taken from the
10 modified file together with unresolved children thereof,
11 if any.

1 21. The system of claim 18, which includes, for an
2 element identified as removed, means for providing the
3 user with the following options:

4 a) do not delete, whereby the merged file has
5 the element as it exists in the base file; and

6 b) delete from the base file, whereby the
7 merged file does not have the element that was deleted
8 from the base file.

1 22. The system of claim 16 wherein the means for
2 providing to user a tree structure comprises means for
3 visually displaying the tree structure.

1 23. The system of claim 22 wherein the means for
2 visually displaying the tree structure comprises means
3 for displaying to the user a screen containing three

1 panes, the first pane displaying said tree structure, the
2 second pane displaying an element of said base file, and
3 the third pane displaying an element of the modified
4 file.

1 24. The system of claim 23 which includes, when the
2 user selects an element of the tree structure displayed
3 in the first pane, means for displaying the source code
4 for the selected element:

5 a) in the second pane if the selected element
6 exists in the base file; and

7 b) in the third pane if the selected element
8 exists in the modified file.

1 25. The system of claim 16 wherein the means for
2 comparing uses an ID attribute of the elements of the
3 base file and the modified file being compared.

1 26. The system of claim 16 wherein the means for
2 comparing uses a name attribute of the elements of the
3 base file and the modified file being compared.

1 27. The system of claim 16 wherein said
2 hierarchically structured files are XML (eXtensible
3 markup language) files and wherein the means for
4 comparing uses:

5 if provided by the elements of the base and
6 modified files being compared, an attribute of type ID;

7 if an attribute of type ID is not provided by
8 the elements of the base and modified files being
9 compared, a <Uuid> tag if provided by the elements of the
10 base and modified files being compared;

1 if an attribute of type ID and a <Uuid> tag is
2 not provided by the elements of the base and modified
3 files being compared, a name attribute if provided by the
4 elements of the base and modified files being compared;
5 and

6 if an attribute of type ID, a <Uuid> tag and a
7 name attribute is not provided by the elements of the
8 base and modified files being compared, a concatenation
9 of a tag of the element and a value of the element.

1 28. The system of claim 16 wherein said
2 hierarchically structured files are XML (eXtensible
3 Markup Language) files.

1 29. A system for determining the differences
2 between two hierarchically structured files comprising:
3 a parser to parse the files and produce a parse
4 tree output for each file; and
5 a comparison module to compare the parse trees
6 output from the parser and to create a merged tree from
7 the parse tree outputs.

1 30. The system of claim 29, further comprising a
2 tree view module to display the merged tree.

[illegible]